



AF/IFW

Approved for use through 07/31/2006. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995, no persons are required to respond to a collection of information unless it displays a valid OMB control number.

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Application Number	10/648,577
Filing Date	8/25/2003
First Named Inventor	Namit Jain
Art Unit	2165
Examiner Name	Radtke, Mark A
Attorney Docket Number	50277-2236

Total Number of Pages in This Submission 21

ENCLOSURES (Check all that apply)

<input type="checkbox"/> Fee Transmittal Form <input checked="" type="checkbox"/> Fee Attached <input type="checkbox"/> Amendment/Reply <input type="checkbox"/> After Final <input type="checkbox"/> Affidavits/declaration(s) <input type="checkbox"/> Extension of Time Request <input type="checkbox"/> Express Abandonment Request <input type="checkbox"/> Information Disclosure Statement <input type="checkbox"/> Certified Copy of Priority Document(s) <input type="checkbox"/> Reply to Missing Parts/Incomplete Application <input type="checkbox"/> Reply to Missing Parts under 37 CFR 1.52 or 1.53	<input type="checkbox"/> Drawing(s) <input type="checkbox"/> Licensing-related Papers <input type="checkbox"/> Petition <input type="checkbox"/> Petition to Convert to a Provisional Application <input type="checkbox"/> Power of Attorney, Revocation <input type="checkbox"/> Change of Correspondence Address <input type="checkbox"/> Terminal Disclaimer <input type="checkbox"/> Request for Refund <input type="checkbox"/> CD, Number of CD(s) _____ <input type="checkbox"/> Landscape Table on CD	<input type="checkbox"/> After Allowance Communication to TC <input type="checkbox"/> Appeal Communication to Board of Appeals and Interferences <input type="checkbox"/> Appeal Communication to TC (Appeal Notice, Brief, Reply Brief) <input type="checkbox"/> Proprietary Information <input type="checkbox"/> Status Letter <input checked="" type="checkbox"/> Other Enclosure(s) (please identify below): Appeal Brief; Check \$500.00 and Return receipt postcard
<div>Remarks</div> <p>The Director is hereby authorized to charge any additional fee(s) or underpayments of fee(s) under 37 CFR 1.16 and 1.17 to Deposit Account Number 50-1302 Deposit Account Name: Hickman Palermo Truong & Becker LLP</p>		

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm Name	Hickman Palermo Truong & Becker LLP		
Signature	<i>Christian A. Nicholes</i>		
Printed name	Christian A. Nicholes		
Date	4/30/2007	Reg. No.	50,266

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below:

Signature	<i>Annette Valdivia</i>		
Typed or printed name	Annette Valdivia	Date	4/30/07

SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450.

MAY 02 2007

PATENT & TRADEMARK OFFICE

PTO/SB/17 (12-04)

Approved for use through 07/31/2008. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

Under the Paperwork Reduction Act of 1995 no persons are required to respond to a collection of information unless it displays a valid OMB control number

Effective on 12/08/2004.

Fees pursuant to the Consolidated Appropriations Act, 2005 (H.R. 4818).

FEE TRANSMITTAL
For FY 2005☐ Applicant claims small entity status. See 37 CFR 1.27TOTAL AMOUNT OF PAYMENT (\$)
500.00**Complete if Known**

Application Number	10/648,577
Filing Date	8/25/2003
First Named Inventor	Namit Jain
Examiner Name	Radtke, Mark A.
Art Unit	2165
Attorney Docket Number	50277-2236

METHOD OF PAYMENT (check all that apply)
☒ Check ☐ Credit Card ☐ Money Order ☐ None ☐ Other (please identify): _____
Deposit Account Deposit Account Number: **50-1302** Deposit Account Name: **Hickman Palermo Truong & Becker LLP**

For the above-identified deposit account, the Director is hereby authorized to: (check all that apply)

☐ Charge fee(s) indicated below ☐ Charge fee(s) indicated below, except for the filing fee
☒ Charge any additional fee(s) or underpayments of fee(s) ☒ Credit any overpayments
under 37 CFR 1.16 and 1.17

WARNING: Information on this form may become public. Credit card information should not be included on this form. Provide credit card information and authorization on PTO-2038.

FEE CALCULATION**1. BASIC FILING, SEARCH, AND EXAMINATION FEES**

Application Type	FILING FEES		SEARCH FEES		EXAMINATION FEES		Fees Paid (\$)
	Fee (\$)	Small Entity Fee (\$)	Fee (\$)	Small Entity Fee (\$)	Fee (\$)	Small Entity Fee (\$)	
Utility	300	150	500	250	200	100	
Design	200	100	100	50	130	65	
Plant	200	100	300	150	160	80	
Reissue	300	150	500	250	600	300	
Provisional	200	100	0	0	0	0	

2. EXCESS CLAIM FEES**Fee Description**

Each claim over 20 (including Reissues)

Each independent claim over 3 (including Reissues)

Multiple dependent claims

Small Entity	
Fee (\$)	Fee (\$)
50	25
200	100
360	180
Multiple Dependent Claims	
Fee (\$)	Fee Paid (\$)

Total Claims	Extra Claims	Fee (\$)	Fee Paid (\$)
_____ - 20 or HP = _____	x _____	= _____	

HP = highest number of total claims paid for, if greater than 20.

Indep. Claims	Extra Claims	Fee (\$)	Fee Paid (\$)
_____ - 3 or HP = _____	x _____	= _____	

HP = highest number of independent claims paid for, if greater than 3

3. APPLICATION SIZE FEE

If the specification and drawings exceed 100 sheets of paper (excluding electronically filed sequence or computer listings under 37 CFR 1.52(e)), the application size fee due is \$250 (\$125 for small entity) for each additional 50 sheets or fraction thereof. See 35 U.S.C. 41(a)(1)(G) and 37 CFR 1.16(s).

Total Sheets	Extra Sheets	Number of each additional 50 or fraction thereof	Fee (\$)	Fees Paid (\$)
_____ - 100 = _____	/ 50 = _____	(round up to a whole number) x _____	= _____	

4. OTHER FEE(S)

Non-English Specification, \$130 fee (no small entity discount)

Other (e.g., late filing surcharge) Appeal Brief

Fees Paid (\$)
\$500.00

SUBMITTED BY

Signature

Registration No.

(Attorney/Agent) **50,266**

Telephone 408.414.1224

Name (Print/Type)

Christian A. Nicholes

Date

4/30/2007

SEND TO: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450. If you need assistance in completing the form, call 1-800-PTO-9199 and select opti



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re application of:

Namit JAIN

Serial No.: 10/648,577

Filed on: August 25, 2003

For: DIRECT LOADING OF OPAQUE TYPES

Confirmation No.: 4090

Examiner: RADTKE, Mark A.

Group Art Unit No.: 2165

MS Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

APPEAL BRIEF

Sir:

This Appeal Brief is submitted in support of the Notice of Appeal filed on February 28, 2007.

I. REAL PARTY IN INTEREST

Oracle International Corporation is the real party in interest.

II. RELATED APPEALS AND INTERFERENCES

The present application is related to U.S. Patent Application Serial No. 10/648,749 and U.S. Patent Application Serial No. 10/648,600, both of which are also under appeal before the Board of Patent Appeals and Interferences. The Honorable Board has not yet handed down a decision in either appeal.

05/03/2007 MWOLDGE1 00000035 10648577

01 FC:1402

500.00 0P

III. STATUS OF CLAIMS

Claims 1-20 have been finally rejected and are the only subjects of this appeal.

IV. STATUS OF AMENDMENTS

The claims were not amended after the Final Office Action.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The present application contains only one independent claim, Claim 1, which is summarized below and annotated to cross-reference features of the claim to specific examples of those features disclosed in the specification. However, the annotations are not intended to limit the scope of the recited features to those specific examples to which the annotations refer.

Claim 1 recites (with added reference annotations in parenthesis) a method of storing data into a database (FIG. 1, database 106), the method comprising:

a loader application (FIG. 1, loader application 102; paragraph [0037]) receiving

(FIG. 2, step 206; paragraphs [0031], [0037], and [0060]) data (FIG. 1, data 118; paragraph [0037]);

determining (FIG. 2, step 208; paragraphs [0031] and [0038]) one or more routines

(FIG. 1, routines 110AA-AN; paragraph [0053]) that are associated

(paragraphs [0030], [0038], and [0059]) with a type (paragraph [0015]) of said

data, wherein said one or more routines are implemented by a program (FIG.

1, opaque type implementor 108A; paragraph [0039]) that is external

(paragraph [0030]) to both said loader application and a database server (FIG.

1, database server 104) that manages said database;

invoking (FIG. 2, step 210; paragraphs [0032], [0040], and [0061]) said one or more routines;

in response (paragraphs [0063]-[0064]) to said one or more routines being invoked, said program performing steps comprising:

creating (FIG. 2, step 212; paragraphs [0032], [0041], and [0063]) a data structure (FIG. 1, array 116) that has one or more elements that correspond (paragraph [0041]) to one or more attributes (paragraph [0015]) of said type; and

populating (FIG. 2, step 214; paragraphs [0032], [0042], and [0064]) said one or more elements with one or more values that are specified (paragraph [0060]) in said data, wherein said one or more values correspond (paragraph [0042]) to said one or more attributes;

generating (FIG. 2, step 216; paragraphs [0033]-[0034], [0044], and [0065]), based on said data structure, a data stream that conforms (paragraph [0044]) to a format of data blocks of said database; and

writing (FIG. 2, step 220; paragraphs [0033]-[0034], [0044], and [0066]-[0068]) said data into one or more data blocks in said database.

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

1. Claims 1-3, 5, 6, 9-11, 13, and 14 stand rejected under 35 U.S.C. § 102(b) as being anticipated, allegedly, by U.S. Patent No. 6,085,198 (“Skinner”).
2. Claims 7 and 15 stand rejected under 35 U.S.C. § 102(b) as being anticipated, allegedly, by Skinner.

3. Claims 8 and 16 stand rejected under 35 U.S.C. § 102(b) as being anticipated, allegedly, by Skinner.

4. Claims 17 and 19 stand rejected under 35 U.S.C. § 102(b) as being anticipated, allegedly, by Skinner.

5. Claims 18 and 20 stand rejected under 35 U.S.C. § 102(b) as being anticipated, allegedly, by Skinner.

6. Claims 4 and 12 stand rejected under 35 U.S.C. § 103(a) as being unpatentable, allegedly, over Skinner in view of Chapter 10, Section 10.1 of “Oracle SQL*Loader: The Definitive Guide” (“O’Reilly”).

VIII. ARGUMENTS

A. The Features of Claims 1-3, 5, 6, 9-11, 13, and 14 Are Not Disclosed, Taught, or Suggested by Skinner

Claim 1 requires that the creation of the “data structure” be performed “in response to” the invocation of “routines” that are “**implemented by a program**” that is **external** to both **the loader application and the database server**. The Examiner’s position is that Skinner’s “client” is the “program” of Claim 1 and Skinner’s “application server” is the “loader application” of Claim 1. Additionally, the Examiner’s position is that the creation of the “data structure,” as recited in Claim 1, is the same as the creation of database tables in the database (as disclosed in Skinner, step 404).

The Examiner analogizes the limitation “creating a data structure that has one or more elements that correspond to said one or more attributes of said type” to what Skinner discloses in col. 16, lines 60-62, and Fig. 4, step 404. This portion of Skinner refers to the creation of **database tables in a database**. Typically, database tables are created in a

database by routines that are **implemented in a database server**. However, Claim 1 requires (a) that this “creating” step be performed by “said program” (“said program performing steps comprising: creating”), (b) that “said program” must be “a program that is **external** to both said loader application **and a database server** that manages said database,” and (c) that the “routines,” in response to whose invocation the “program” performs the “creating,” must be **“implemented by”** that same “program.” Even if Skinner’s approach creates database tables in a database, Skinner does **not** disclose, teach, or suggest that these database tables are created **by a program that is external to a database server** or in response to the invocation of routines that are **implemented by** such an external program. Therefore, Skinner does not disclose the “creating” step of Claim 1 being performed by “a program that is **external** to both said loader application **and a database server** that manages said database” as required by Claim 1.

The Examiner notes that Skinner’s step 402 involves creating data class definitions **for** an application tier and a client tier, apparently reasoning that the application tier and the client tier are external to a database server. However, even if the data class definitions are created **for** these tiers, Skinner does not disclose, teach, or suggest that these data class definitions are created **by** these tiers. Additionally, it appears to be the Examiner’s position that the **database tables** of Skinner’s step 404, rather than the data class definitions discussed in Skinner’s step 402, are analogous to the “data structure” recited in Claim 1.

Claim 1 also requires that the “data stream” be “generated based on said data structure.” The Examiner’s position is that “creating a data structure,” as recited in Claim 1, is the same as the creation of the database tables in the database (as disclosed in Skinner, step 404). However, this position does not make sense, because if the “data structure” of Claim 1

were a database table that was created in the database and then populated with values, then there would be no need to also “generate, based on” the database table, a data stream that conformed to a format of data blocks of the database. Skinner doesn't disclose that any “data stream,” which “conforms to a format of the data blocks of the database,” is “generated based on” a database table (the alleged “data structure”) that has already been created in the database. Therefore, Skinner does not disclose, teach, or suggest that a “data stream” is “generated based on said data structure” as required by Claim 1.

For at least the above reasons, Claim 1 is patentable over Skinner under 35 U.S.C. § 102(b).

By virtue of their dependence from Claim 1, Claims 2, 3, 5, 6, 9-11, 13, and 14 include the features of Claim 1 that are distinguished from Skinner above. As a result, Claims 2, 3, 5, 6, 9-11, 13, and 14 are patentable over Skinner under 35 U.S.C. § 102(b) for at least the reasons discussed above in connection with Claim 1. The rejection of Claims 1-3, 5, 6, 9-11, 13, and 14 should be reversed.

B. The Features of Claims 7 and 15 Are Not Disclosed, Taught, or Suggested by Skinner

Claim 7 depends from Claim 1, and further recites “adding, to a table, an entry that indicates an association between said type and said one or more routines.” For example, such a table is discussed in paragraph [0038] of the present application, which provides an example of dispatch table 112. The addition of such an entry to the table is discussed in paragraph [0039] of the present application, for example.

The Examiner alleges that Skinner discloses this feature in col. 19, lines 66-67, and col. 20, lines 15-19. The Examiner appears to allege that the vector of “MetaMethod”

instances discussed in these sections of Skinner is analogous to the “table” of Claim 7. The Examiner appears to allege that the “methods” described by the “MetaMethod” instances are analogous to the “routines” of Claim 7.

However, even if Skinner’s vector of “MetaMethod” instances is interpreted as associating a “type” with a “routine,” Skinner does **not** disclose that any of these “MetaMethod” instances describe routines **in response to whose invocation** the program that **implements** those routines (a) **creates** a data structure that has elements that correspond to attributes of the type; and (b) **populates** the elements with attribute-correspondent values that are specified in the data. It should be noted that the “routines” recited in Claim 7 must be the **same** “routines” that are recited in Claim 1. Thus, the “routines” recited in Claim 7 must have all of the characteristics of the “routines” recited in Claim 1. The Examiner is not permitted to swap out one “routine” for another entirely different “routine” in between the rejections of Claims 1 and 7.

The methods described by the “MetaMethod” instances, when invoked, do not cause any program that implements those methods to create and populate the data structure that is recited in Claim 1. As is discussed above with reference to Claim 1, the Examiner apparently analogizes the “data structure” of Claim 1 to **database tables**. However, the methods described by the “MetaMethod” instances **do not** cause any program that implements those methods to **create** or **populate** such database tables.

Even if the “data structures” of Claim 1 are taken to be analogous to the “data class definitions” whose creation is discussed with reference to Skinner’s step 402, Skinner still does **not** disclose, teach, or suggest that such “data class definitions” are created by a

program that implements the methods described by the “MetaMethod” instances in response to the invocation of those methods.

In summary, even if Skinner’s vector of “MetaMethod” instances actually does associate a “type” with “one or more routines,” those alleged “one or more routines” still **do not** have the characteristics that are required by Claim 1, and therefore also by Claim 7. The methods described by the vector’s “MetaMethod” instances **do not** create or populate any of the things that the Examiner analogized to the “data structures” of Claim 1. Therefore, Skinner does not disclose, teach, or suggest “adding, to a table, an entry that indicates an association between said type and **said** one or more routines” as recited in Claim 7.

For at least the above reasons, Claim 7 is patentable over Skinner under 35 U.S.C. § 102(b).

By virtue of its dependence from Claim 7, Claim 15 includes the features of Claim 7 that are distinguished from Skinner above. As a result, Claim 15 is patentable over Skinner under 35 U.S.C. § 102(b) for at least the reasons discussed above in connection with Claim 7. The rejection of Claims 7 and 15 should be reversed.

C. The Features of Claims 8 and 16 Are Not Disclosed, Taught, or Suggested by Skinner

Claim 8 depends from Claim 1, and further recites “wherein said invoking comprises invoking one or more routines that are located at one or more **addresses that are associated with said type via an associative structure.**” For example, such an associative structure is discussed in paragraph [0038] of the present application, which provides an example of dispatch table 112. Dispatch table 112 associates a type with addresses of routines.

The Examiner alleges that Skinner discloses this feature in col. 18, lines 6-10. The Examiner appears to allege that the “methods” discussed in this section of Skinner are analogous to the “routines” that are recited in Claim 8. However, even assuming that such “methods” are analogous to the “routines” of Claim 8, this section of Skinner does not say anything about the **addresses** of such “methods” being “associated, via **an associative structure,**” with a “**type.**” This section of Skinner says nothing about “addresses,” an “associative structure,” or any “type.” Therefore, Skinner does not disclose, teach, or suggest “wherein said invoking comprises invoking one or more routines that are located at one or more **addresses that are associated with said type via an associative structure**” as recited in Claim 8.

For at least the above reasons, Claim 8 is patentable over Skinner under 35 U.S.C. § 102(b).

By virtue of its dependence from Claim 8, Claim 16 includes the features of Claim 8 that are distinguished from Skinner above. As a result, Claim 16 is patentable over Skinner under 35 U.S.C. § 102(b) for at least the reasons discussed above in connection with Claim 8. The rejection of Claims 8 and 16 should be reversed.

D. The Features of Claims 17 and 19 Are Not Disclosed, Taught, or Suggested by Skinner

Claim 17 depends from Claim 1, and further recites “said program registering, with said loader application, said one or more routines, which are not implemented by said loader application.” For example, such a registration is discussed in paragraph [0030] of the present application.

The Examiner alleges that Skinner discloses this feature in columns 32-35 and in column 24, lines 12-45. These portions of Skinner say nothing about registering, with a loader application, routines that are not implemented by the loader application.

The Examiner alleges that “routines not implemented by the loader application are routines **of** opaque types.” This is simply not correct. Paragraph [0021] of the present application, to which the Examiner refers, states, “According to one embodiment of the present invention, a program that implements an opaque type (an “opaque type implementor”) registers, with a loader application, one or more routines that are associated with the opaque type. The opaque type implementor, which is external to both the loader application and a database server that manages the database, implements the routines.” There is nothing in paragraph [0021], or anywhere else in the present application, that indicates that the routines are **of** or **within** or a **part of** any opaque type.

Instead, the present application discloses an embodiment in which the “routines,” when invoked, cause the “opaque type implementor” to create and populate a “data structure” (e.g., array 116) that corresponds to the opaque type. Because the recited “routines” are **not a part of** the opaque type, the Examiner’s discussion of “reflection methods” is completely irrelevant to the patentability of Claim 17. Skinner does not disclose, teach, or suggest “said program registering, with said loader application, said one or more routines, which are not implemented by said loader application” as recited in Claim 17.

Additionally, the “routines” recited in Claim 17 are the **same** “routines” as those recited in Claim 1, and therefore must have the same characteristics are those recited in Claim 1. Skinner does not disclose that a program registers, with a loader application,

routines having such characteristics (e.g., routines that create and populate a data structure as discussed above with reference to Claim 1).

For at least the above reasons, Claim 17 is patentable over Skinner under 35 U.S.C. § 102(b).

By virtue of its dependence from Claim 17, Claim 19 includes the features of Claim 17 that are distinguished from Skinner above. As a result, Claim 19 is patentable over Skinner under 35 U.S.C. § 102(b) for at least the reasons discussed above in connection with Claim 17. The rejection of Claims 17 and 19 should be reversed.

E. The Features of Claims 18 and 20 Are Not Disclosed, Taught, or Suggested by Skinner

Claim 18 depends from Claim 1, and further recites “said loader application invoking at least one of said one or more routines to find out (a) a number of one or more attributes within an opaque type and (b) one or more types of said one or more attributes within said opaque type.” As is clear from Claim 1, the “said one or more routines” must be implemented by the recited “program.” As is discussed above with reference to Claim 1, the Examiner apparently analogizes the “program” to Skinner’s “client.” However, the “reflection methods” which the Examiner analogizes to the “said one or more routines” in the rejection of Claim 18 are **not** implemented by Skinner’s “client.”

Additionally, the “routines” recited in Claim 18 are the **same** “routines” as those recited in Claim 1, and therefore must have the same characteristics are those recited in Claim 1. Skinner does not disclose that the “reflection methods” that the Examiner analogizes to the “said one or more routines” in the rejection of Claim 18 have such

characteristics (e.g., routines that create and populate a data structure as discussed above with reference to Claim 1).

For at least the above reasons, Claim 18 is patentable over Skinner under 35 U.S.C. § 102(b).

By virtue of its dependence from Claim 18, Claim 20 includes the features of Claim 18 that are distinguished from Skinner above. As a result, Claim 20 is patentable over Skinner under 35 U.S.C. § 102(b) for at least the reasons discussed above in connection with Claim 18. The rejection of Claims 18 and 20 should be reversed.

F. The Rejection of Claims 4 and 12 is Based Entirely on Hindsight Reasoning.

As is discussed above, Claim 1 recites features that Skinner, by itself, does not disclose, teach or suggest. The Examiner does not allege that O'Reilly discloses these features. Therefore, even if Skinner and O'Reilly could be combined, the combination still would not disclose all of the features of Claim 1. By virtue of their dependence from Claim 1, Claims 4 and 12 inherit the distinguished features of Claim 1. Therefore, Claims 4 and 12 are patentable over Skinner and O'Reilly, taken individually or in combination, under 35 U.S.C. § 103(a).

Additionally, the Examiner's alleged motivation to combine Skinner and O'Reilly is based entirely on impermissible hindsight reasoning. The Appellants admit that direct path loading, in general, was known prior to the filing of the present application. Indeed, the present application discusses direct path loading in its own "background" section (paragraph [0017]). The Examiner uses O'Reilly to allege nothing more than what the Appellants have already admitted. However, at the time of Skinner and O'Reilly, people of ordinary skill in

the art did not know how to load data of **opaque types**, whose structure was not known to the loader, into a database using the direct path loading approach. Using the method of Claim 4, routines implemented by an opaque type implementor may be invoked by the loader application in order to enable the loader application to load opaque type data into a database using the direct path loading approach. Even if O'Reilly discloses direct path loading generally, O'Reilly does not contemplate the direct path loading of data that is of opaque types. Even given the teachings of Skinner and O'Reilly, a person of ordinary skill in the art at the time that the present application was filed would not have known how to load data of opaque types into a database using a direct path loading approach.

Therefore, the rejection of Claims 4 and 12 under 35 U.S.C. § 103(a) is impermissibly based on hindsight reasoning. The rejection of Claims 4 and 12 should be reversed.

IX. CONCLUSION AND PRAYER FOR RELIEF

Based on the foregoing, it is respectfully submitted that the rejections of Claims 1-20 lack the requisite factual and legal bases. Appellants respectfully request that the Honorable Board **reverse** the rejections of Claims 1-20.

Respectfully submitted,

HICKMAN PALERMO TRUONG & BECKER LLP



Christian A. Nicholes
Registration No. 50,266

Date: April 30, 2007

2055 Gateway Place, Suite 550
San Jose, California 95110-1089
Tel: (408) 414-1224
Fax: (408) 414-1076

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service as first class mail in an envelope addressed to: MS Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450

on 4/30/07 by 
Annette Valdivia

CLAIMS APPENDIX

- 1 1. A method of storing data into a database, the method comprising:
2 a loader application receiving data;
3 determining one or more routines that are associated with a type of said data, wherein
4 said one or more routines are implemented by a program that is external to
5 both said loader application and a database server that manages said database;
6 invoking said one or more routines;
7 in response to said one or more routines being invoked, said program performing
8 steps comprising:
9 creating a data structure that has one or more elements that correspond to one
10 or more attributes of said type; and
11 populating said one or more elements with one or more values that are
12 specified in said data, wherein said one or more values correspond to
13 said one or more attributes;
14 generating, based on said data structure, a data stream that conforms to a format of
15 data blocks of said database; and
16 writing said data into one or more data blocks in said database.
- 1 2. The method of Claim 1, wherein a number of attributes of said type is not defined to
2 said loader application.
- 1 3. The method of Claim 1, wherein a type of an attribute of said type of said data is not
2 defined to said loader application.

- 1 4. The method of Claim 1, wherein said creating, said populating, said generating, and
2 said writing are performed using a direct path loading approach rather than a
3 conventional path loading approach, and without causing a Structured Query
4 Language (SQL) engine to load said data.
- 1 5. The method of Claim 1, wherein said data structure is created, by said program, in a
2 memory space of said loader application rather than a memory space of said program.
- 1 6. The method of Claim 1, wherein said determining comprises locating addresses of
2 one or more routines that are in a same entry of a table as an identity of said type.
- 1 7. The method of Claim 1, further comprising:
2 adding, to a table, an entry that indicates an association between said type and said
3 one or more routines.
- 1 8. The method of Claim 1, wherein said invoking comprises invoking one or more
2 routines that are located at one or more addresses that are associated with said type
3 via an associative structure.
- 1 9. A computer-readable storage medium carrying one or more sequences of instructions
2 which, when executed by one or more processors, causes the one or more processors
3 to perform the method recited in Claim 1.
- 1 10. A computer-readable storage medium carrying one or more sequences of instructions
2 which, when executed by one or more processors, causes the one or more processors
3 to perform the method recited in Claim 2.

1 11. A computer-readable storage medium carrying one or more sequences of instructions
2 which, when executed by one or more processors, causes the one or more processors
3 to perform the method recited in Claim 3.

1 12. A computer-readable storage medium carrying one or more sequences of instructions
2 which, when executed by one or more processors, causes the one or more processors
3 to perform the method recited in Claim 4.

1 13. A computer-readable storage medium carrying one or more sequences of instructions
2 which, when executed by one or more processors, causes the one or more processors
3 to perform the method recited in Claim 5.

1 14. A computer-readable storage medium carrying one or more sequences of instructions
2 which, when executed by one or more processors, causes the one or more processors
3 to perform the method recited in Claim 6.

1 15. A computer-readable storage medium carrying one or more sequences of instructions
2 which, when executed by one or more processors, causes the one or more processors
3 to perform the method recited in Claim 7.

1 16. A computer-readable storage medium carrying one or more sequences of instructions
2 which, when executed by one or more processors, causes the one or more processors
3 to perform the method recited in Claim 8.

1 17. The method of Claim 1, further comprising:
2 said program registering, with said loader application, said one or more routines,
3 which are not implemented by said loader application; and
4 in response to said program registering said one or more routines with said loader
5 application, said loader application adding, to a dispatch table, an entry that
6 indicates an association between said one or more routines and an opaque type
7 implemented by said program.

1 18. The method of Claim 1, wherein invoking said one or more routines comprises:
2 said loader application invoking at least one of said one or more routines to find out
3 (a) a number of one or more attributes within an opaque type and (b) one or
4 more types of said one or more attributes within said opaque type; and
5 said loader application invoking at least one of said one or more routines to populate,
6 with values of instances of the opaque type, elements of an array that is stored
7 in a memory space of said loader application.

1 19. A computer-readable storage medium carrying one or more sequences of instructions
2 which, when executed by one or more processors, causes the one or more processors
3 to perform the method recited in Claim 17.

1 20. A computer-readable storage medium carrying one or more sequences of instructions
2 which, when executed by one or more processors, causes the one or more processors
3 to perform the method recited in Claim 18.

EVIDENCE APPENDIX

None.

RELATED PROCEEDINGS APPENDIX

No available decisions.